

EXHIBIT 14

Filed Under Seal

HIGHLY CONFIDENTIAL - SOURCE CODE - ATTORNEYS' EYES ONLY

1 Google's first infringement.

2 **B. Alleged Alternative #2**

3 478. Google argues that its alleged “non-infringing alternative” #2— “to have the sender
4 device maintain the queue locally on the sender device and configure the sender device to playback
5 the local queue” — is relevant to limitation 1.4 of claim 1 of the '033 Patent that requires “operating
6 in a first mode in which the computing device is configured for playback of *a remote playback*
7 *queue* provided by a cloud-based computing system associated with a cloud-based media service.”
8 *See id.*

9 479. It is my opinion that there are various flaws to Google's assertions regarding alleged
10 “non-infringing alternative” #2.

11 480. **First**, Google has not provided sufficient details regarding this alleged alternative.
12 For example, Google has not provided sufficient details as to what is meant by “the sender device
13 *maintain[s]* the queue locally on the sender device” and how that necessarily results in limitation
14 1.4 of claim 1 not being practiced. For instance, just because a sender device might *maintain* a
15 local copy of a queue does not necessarily mean that there is not also “a remote playback queue
16 *provided by* a cloud-based computing system associated with a cloud-based media service.” This
17 appears to be Google once again pushing the false premise that there can only be one “playback
18 queue” in a system. I disagree with their premise for many of the same reasons that I explained at
19 paragraphs 276-282 of my Opening Expert Report for the '615 Patent, which I incorporate by
20 reference in their entirety. Opening Expert Report of Douglas C. Schmidt (June 22, 2022). As
21 another example, Google has not provided sufficient details as to how the Sender would transfer
22 playback responsibility for the “local queue” to a Receiver in this alleged alternative. As a result,
23 I do not have enough information to fully evaluate whether this alleged alternative would have been
24 non-infringing, available, technically feasible, or commercially acceptable. Nevertheless, I have
25 made my best effort to respond to this alleged alternative based on my current understanding of the
26 limited information provided by Google. I expressly reserve the right to supplement my opinions
27 regarding this alleged alternative if and when Google provides sufficient details.

28 481. **Second**, I have seen no evidence to support Google's assertion that “it would need

HIGHLY CONFIDENTIAL - SOURCE CODE - ATTORNEYS' EYES ONLY

1 the GetWatchNext and GetPlayer services would be changed. *See* Google LLC's Eighth
2 Supplemental Objections and Responses to Plaintiff Sonos Inc.'s First Set of Fact Discovery
3 Interrogatories (No. 18), p. 13 ("[T]he receiver device would send a request to a Onesie agent. The
4 Onesie agent would send the GetWatchNext request and GetPlayer request, and then stream the
5 media data from the Onesie agent to the receiver device."). Moreover, Google has provided no
6 details as to what it means for the "Streaming Watch" feature to "allow the agent to make the
7 GetWatchNext and GetPlayer calls." *Id.*, 13-14. Such information is critical to make a
8 determination of whether Google's proposed alleged alternative would still infringe, at a minimum,
9 under DoE.

10 502. For example, while Google's interrogatory response broadly incorporates by
11 reference Dr. Bhattacharjee's patent showdown expert report, Dr. Bhattacharjee stated "a Bandid
12 server (called the 'Onesie' agent) would provide the receiver device with both the Bandid URL
13 and the chunks of media content." Rebuttal Expert Report of Samrat Bhattacharjee Regarding Non-
14 Infringement of Claim 13 of U.S. Patent No. 9,967,615 and Other Issues, (July 27, 2022), ¶359.
15 Assuming that Dr. Bhattacharjee's articulation is accurate, it appears to me that the Receiver would
16 still communicate (via the aforementioned "request") with the YouTube cloud infrastructure
17 ("Onesie" agent) to obtain data (e.g., Bandid URL) identifying a next one or more media items
18 that are in the remote playback queue and the Receiver would still use the obtained data (e.g.,
19 Bandid URL) to retrieve at least one media item in the remote playback queue (e.g., certain
20 "chunks of media content"). *See also, e.g.*, GOOG-SONOSNDCA-00073494, 95 ("Onesie only
21 streams the first seven seconds of a video. For the rest of the video, the Player needs to fetch it from
22 the content-hosting machine.").

23 503. As another example, while Google's interrogatory response vaguely refers to a
24 "Streaming Watch" feature, the cited documents and Dr. Bhattacharjee's reports are vague and
25 ambiguous on this point and seem to suggest that the GetPlayer call would be for the current media
26 item and the simultaneous GetWatchNext call would be for the next media item. If so, it appears
27 to me that the Receiver would still communicate (via a first "Streaming Watch" "request") with the
28 YouTube cloud infrastructure ("Onesie" agent) to obtain data identifying a next one or more media